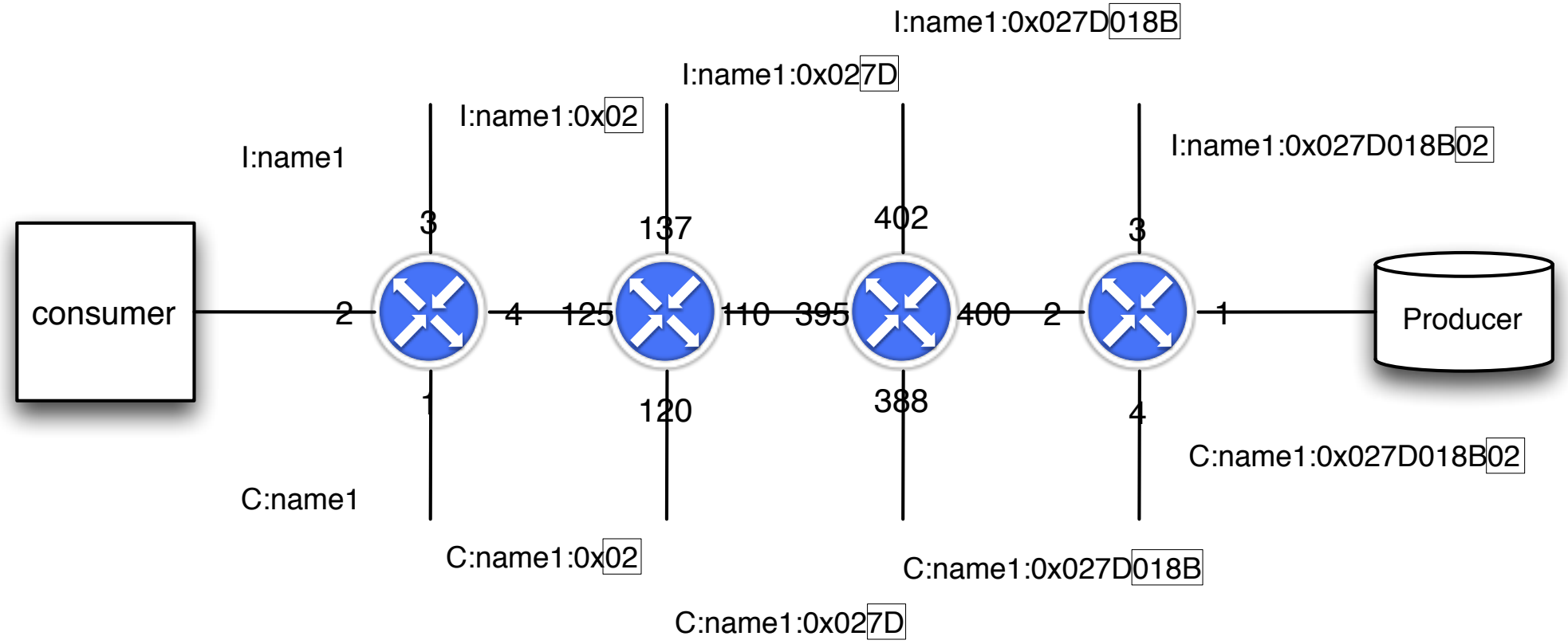
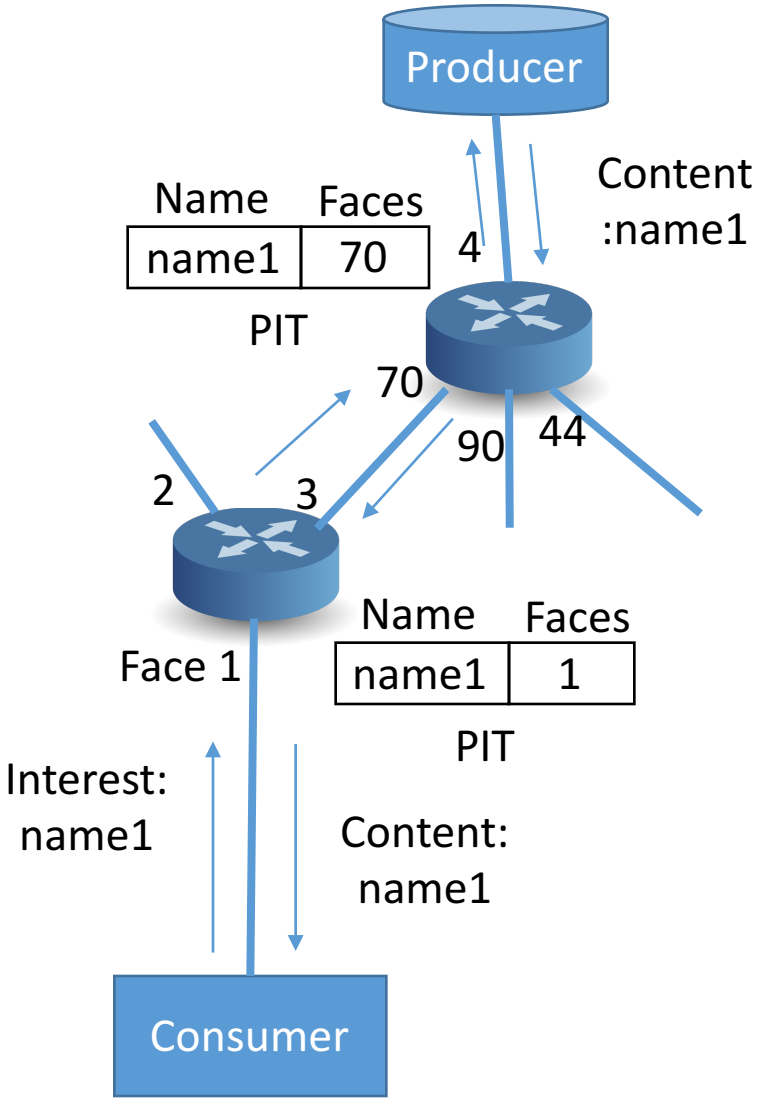
Face-Stack CCN Forwarding for Trusted Domains

Core Concept

- Instead of the forwarding state existing in local memory in the forwarder's PIT, the state is accumulated in the Interest message, it is then transferred to the Content Object message by the Producer, and stripped off enroute back to the Consumer

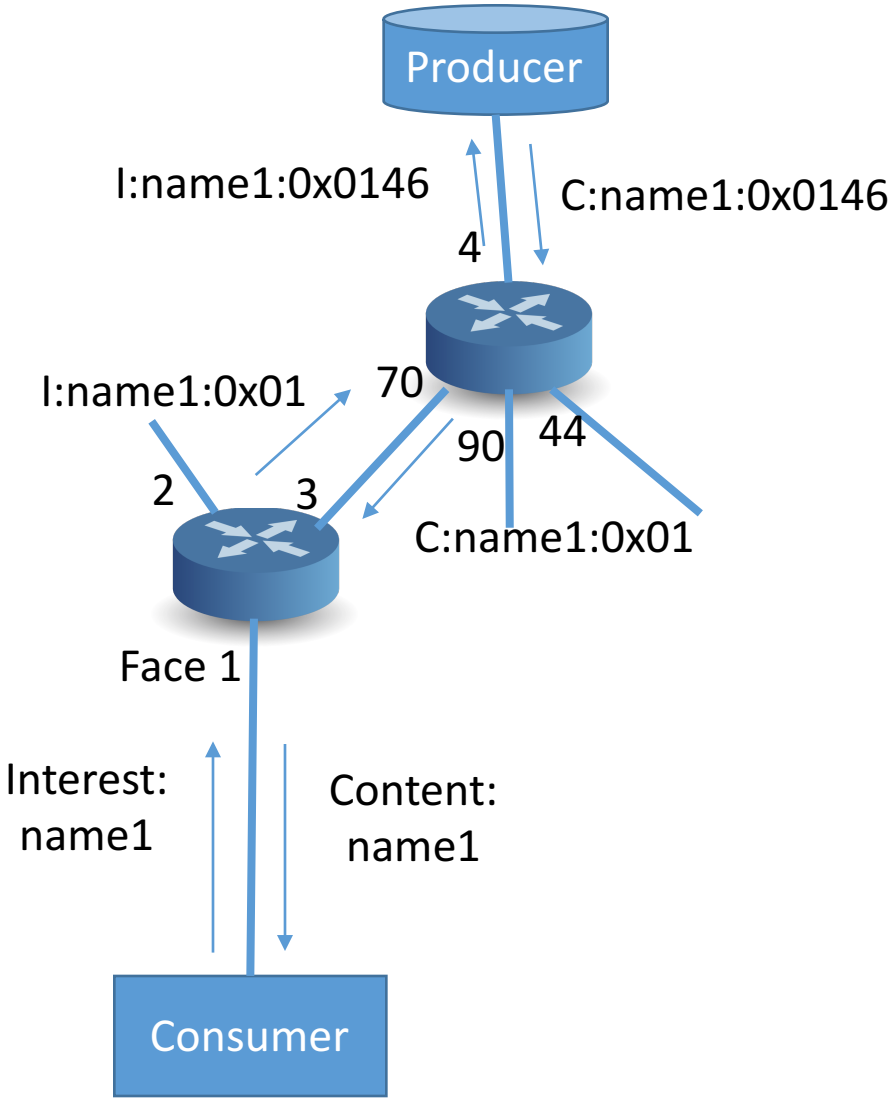


ICN Router Operation



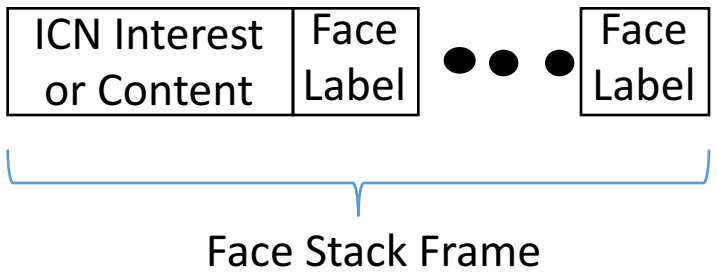
- A data consumer sends an Interest for some named data, e.g. ”name1”
- An ICN router records the pending Interest and its source “face” in Pending Interest Table (PIT), and forwards the Interest
- A producer responds to the Interest with the Content corresponding to the Interest name.
- Each router forwards the Content on the face(s) for the pending Interest recorded in its PIT. The PIT entry for that name is removed.
- The number of PIT entries will be a function of the incoming interest rate and the round trip time for a corresponding content object, e.g. 40 thousand entries assuming 1,250 byte interests, 10 gigabit link and average 40 msec round trip time.

Face Stack Forwarding ICN Router Operation



- A data consumer sends an Interest for some named data, e.g. ”name1”
- An ICN router appends the source “face” to the Interest, and forwards the Interest
- A producer responds to the Interest with the Content corresponding to the Interest name, and appends the face stack.
- Each router forwards the Content based on the outermost face label, removing that label before forwarding.
- There are no Pending Interest Tables.
- Each Interest and Content increases by the maximum size of the face stack, e.g. average of 10 bytes for an average of 10 hops and 1 byte face label.

Face Stack Forwarding Frame



- A face stack frame consists of an ICN Interest or Content frame and one or more face labels.
- A face label is one or more bytes. Each forwarder uses a fixed size face label.
- A forwarder will distinguish the use of a face stack frame from a regular ICN frame based on the layer 2 connection type.

Definitions

- Face Label
  - N*8 bit value
  - N is an integer constant that is defined locally in each forwarder
    - No need to communicate or coordinate with other forwarders
    - N should be as small as is practical for each forwarder
  - Unique (and constant) value for each Face within each forwarder
    - Only locally unique, no need to communicate or coordinate with other forwarders
- Face Stack
  - List of Face Labels appended to the Interest and Content Object messages
    - A growing set of bytes as Interest is forwarded through the network
    - A diminishing set of bytes as Content Object is forwarded
  - Opaque, except for final N*8 bits that are locally meaningful to each forwarder
- Face Stack Router (FSR)
  - A CCN forwarder that supports Face Stack Forwarding

FSR Forwarder Behavior - Interest Processing

- CS lookup – identical to normal CCN forwarder
- FIB lookup
  - Append ingress Face ID to the Face Stack
  - If the egress face is a link to a forwarder that doesn't support FSR then create PIT entry, storing the accumulated Face Stack.
  - If the egress face is a link to an untrusted domain (3rd party, interconnection link, peering link, customer domain) that supports FSR, the Face Stack can be left intact, but obfuscated, and no PIT entry needed.
  - If the egress face is a link to an FSR within the trusted domain, simply forward, with no PIT entry created.

Forwarder Behavior – Content Object Processing

- If the ingress face is a link from an FSR within the trusted domain, pop the final label off the CO and forward out the egress face with the matching ID. Cache the CO if cacheable.
- If the ingress face is a link from an FSR in an untrusted domain, de-obfuscate the face stack, pop the final label off the CO and forward out the egress face with the matching ID. Cache the CO if cacheable.
- If the ingress face is a link to a non-FSR, do PIT lookup using CO name to recover face stack. pop the final label off the CO and forward out the egress face with the matching ID. Cache the CO if cacheable.

Consumer Behavior

- No change from normal CCN consumer

Producer Behavior

- Face Stack aware producer MUST copy face stack from Interest message onto returned Content Object

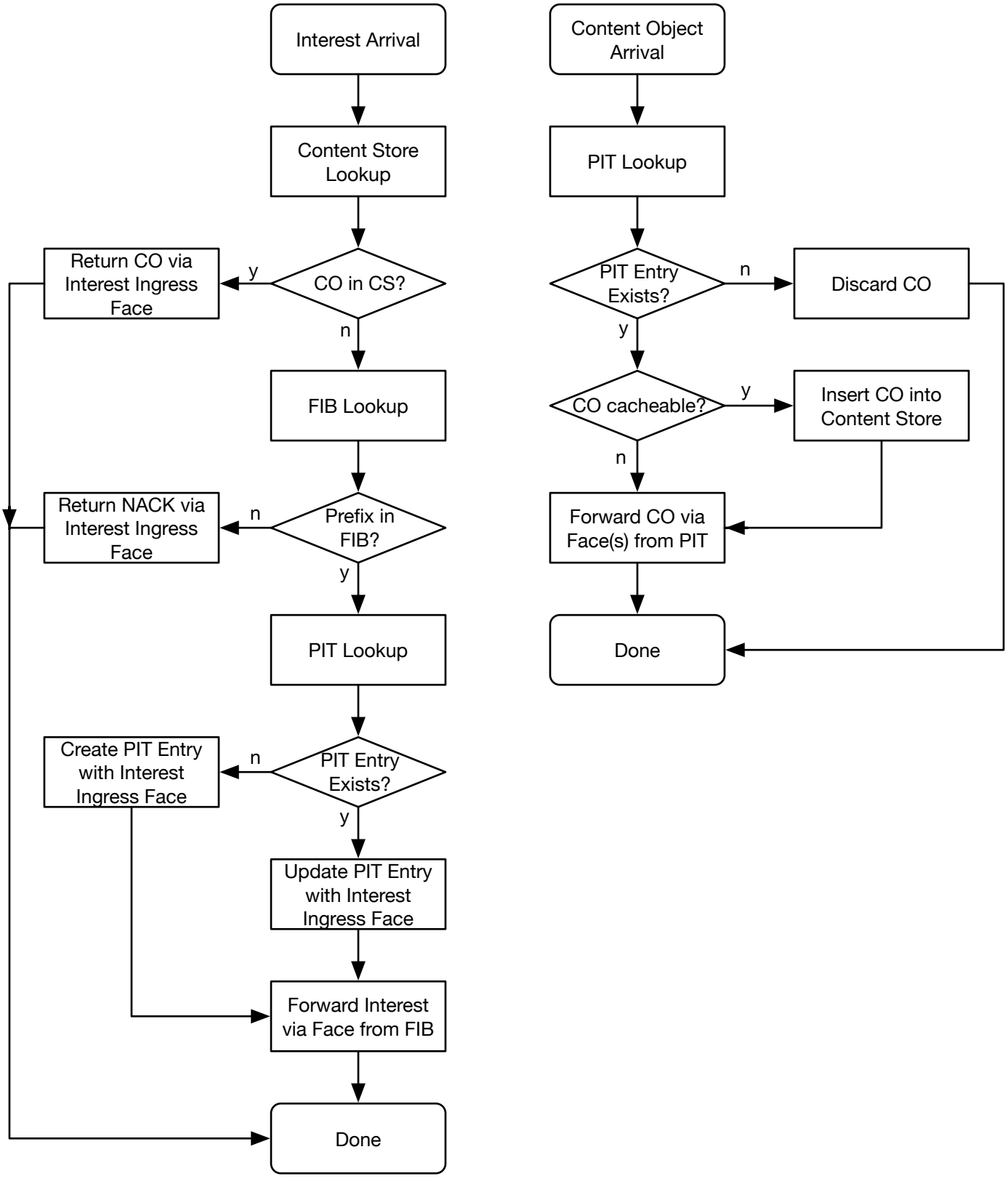
Benefits

- Simpler, higher performance, cheaper core routers
  - For routers within the trusted domain, no PIT required.
- New communication modalities
  - Enables unsolicited communication to consumers within the trusted domain
    - Application instance can send an Interest to ”register” itself with a producer.
    - Producer stores the label stack associated with the application instance
    - Producer can then send unsolicited Content Objects to the application, by using the label stack
  - Content Pre-positioning
- Better Network Management
  - At any location in a network, label stack identifies a consumer for accounting, rate limiting, network protection, etc.
- Reduces attack surface
  - Interest flooding attack can’t target a PIT that doesn’t exist!

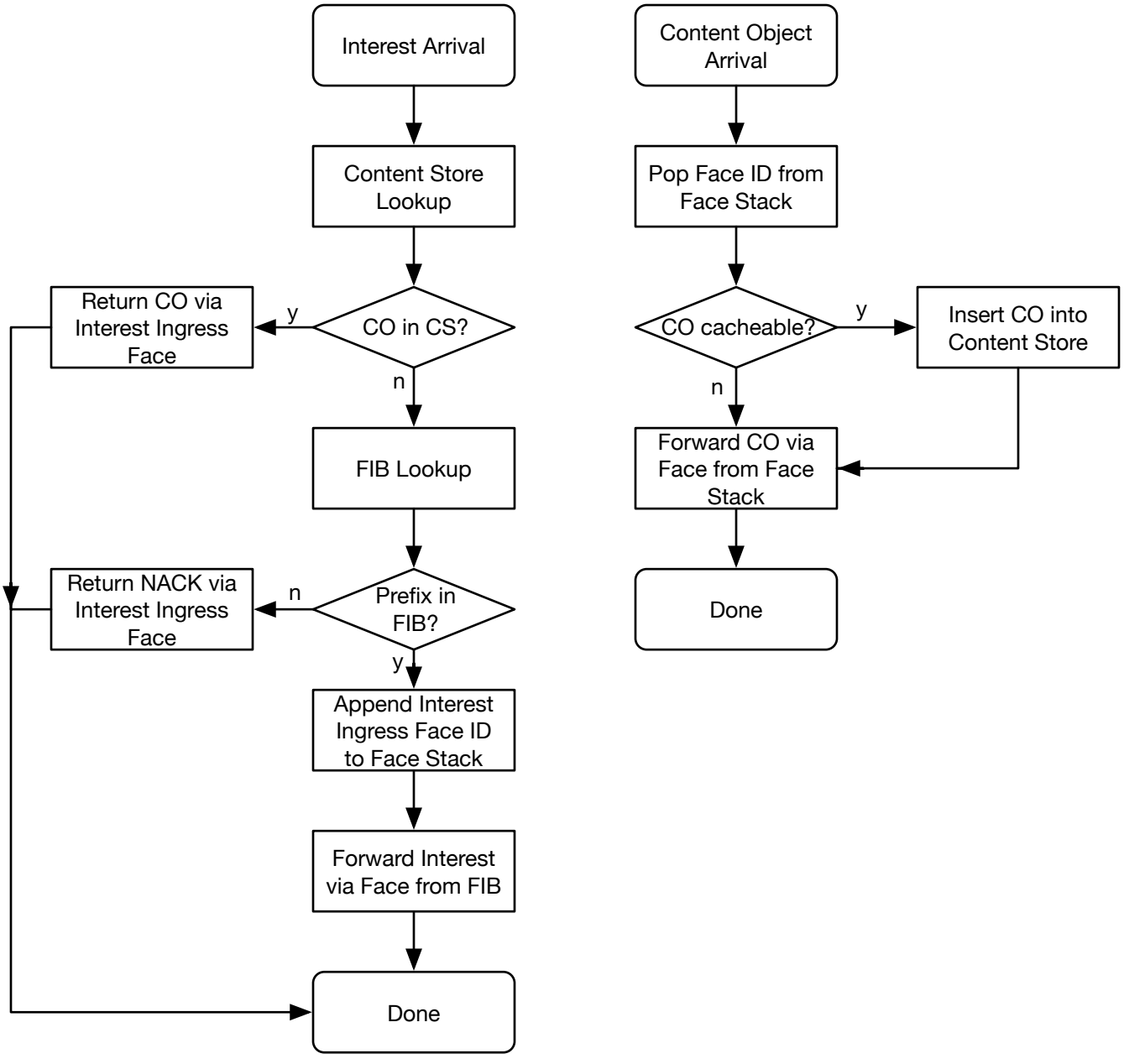
Drawbacks

- Increased PIT memory requirement for border routers
  - Need to store face stack instead of just Ingress Face ID
- Propagation of user-identifying information through the network
  - The accumulated face stack is unique to the user, potentially facilitating correlation of requests to users by in-network bulk traffic collection equipment
- Negligible reduction in “multicast” benefit of CCN

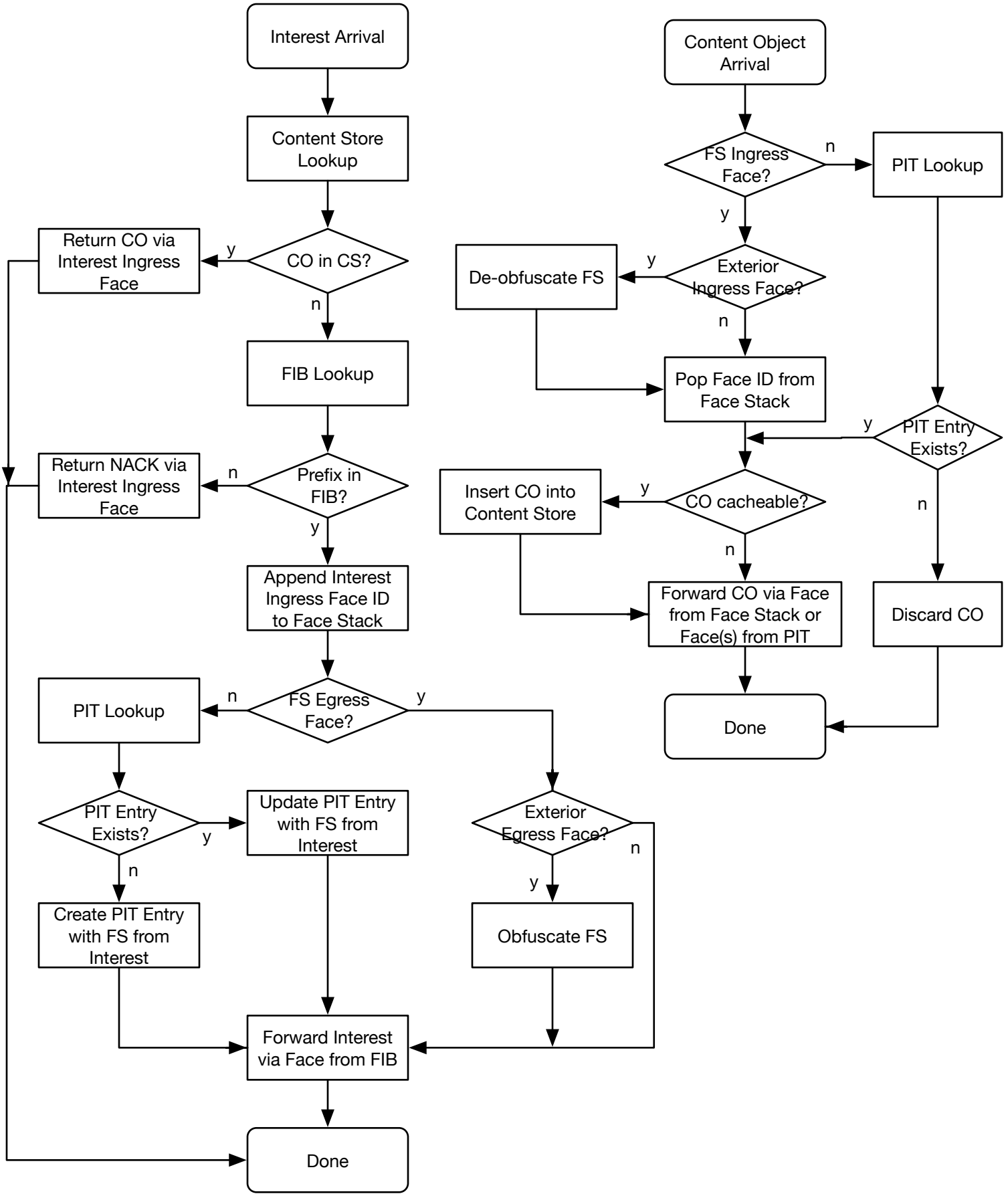
Traditional CCN Forwarder



Interior CCN Face Stack Router



Border CCN Face Stack Router



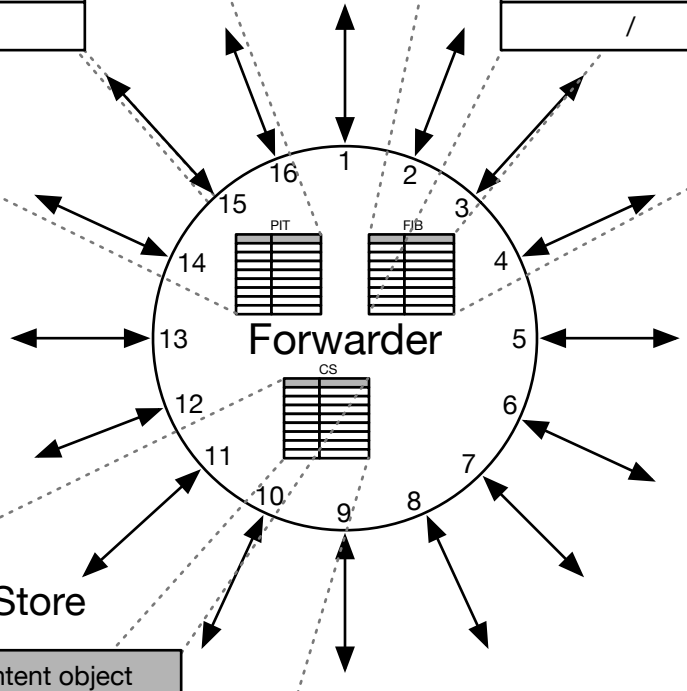
Traditional CCN Forwarder

Pending Interest Table

hash of name	face
9A723BF5	2
9DF87635	4
7BCCFE10	12
2D34F29F	5,7
1114BF78	6
B76AD468	3
...	...
5EC66BDD	3

Forwarding Information Base

prefix	face
/isp/mail/	3
/isp/database/	2
/isp/video/east/	13
/isp/video/west	5
/isp/web/	2,13
/isp/netmon/	3
...	...
/	1



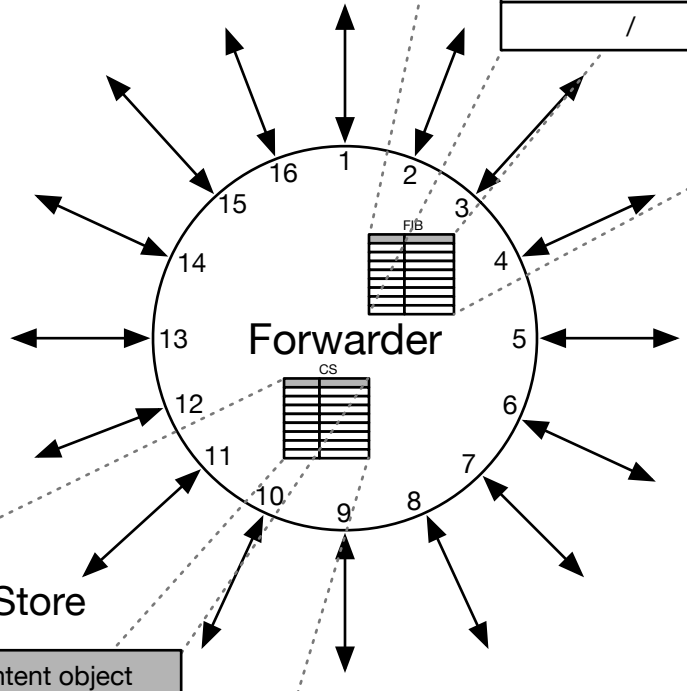
Content Store

hash of name	content object
83D6F0CC	<CO 1>
B051543C	<CO 2>
69831BE6	<CO 3>
8DE7621F	<CO 4>
3BA2A312	<CO 5>
BFC9CBA5	<CO 6>
...	...
B9872680	<CO N>

Interior CCN Face Stack Router

Forwarding Information Base

prefix	face
/isp/mail/	3
/isp/database/	2
/isp/video/east/	13
/isp/video/west	5
/isp/web/	2,13
/isp/netmon/	3
...	...
/	1



Content Store

hash of name	content object
83D6F0CC	<CO 1>
B051543C	<CO 2>
69831BE6	<CO 3>
8DE7621F	<CO 4>
3BA2A312	<CO 5>
BFC9CBA5	<CO 6>
...	...
B9872680	<CO N>